App. No. 09/963,270
RCE to FOA of 02/22/06

Amendments to the Claims

This listing of claims will replace all prior versions, and listings, of the claims:

1. (currently amended) A method for providing hint instructions to a processor, comprising:

generating hint code that includes (i) a hint instruction in response to a set of object code to be executed by the processor and (ii) a selected instruction to be removed from the set of object code;

replacing the selected instruction in the object code with a break instruction~~inserting a break instruction in place of the selected instruction in the object code~~ such that the break instruction causes the processor to obtain and execute both the hint instruction and the selected instruction; and

resuming execution of the object code without changing addresses of subsequent instructions in the object code.

2. (previously presented) The method of claim 1, wherein the hint code further includes (iii) an instruction for the processor to resume execution of the set of object code.

3. (previously presented) The method of claim 1, wherein the hint code adapts the set of object code so the set of object code can be executed by the processor.

4. (previously presented) The method of claim 1, further comprising loading the hint instruction into a hint register such that the break instruction causes the processor to obtain the hint instruction from the hint register and execute the hint instruction.

5. (original) The method of claim 4, wherein the step of loading the hint instruction into a hint register further includes the step of loading the selected instruction into the hint register such that the break instruction causes the processor to obtain the selected instruction from the hint register and execute the selected instruction.

6. (original) The method of claim 4, wherein the step of loading the hint instruction into a hint register further includes the step of loading an address into the hint register such that the break instruction causes the processor to load the hint register using the address.

7. (previously presented) The method of claim 1, further comprising determining the hint instruction in response to a micro-architecture of the processor.

8. (previously presented) A computer system, comprising:

object code adapter that determines a hint instruction that includes a branch prediction instruction in response to a set of object code and that inserts a break instruction into the object code, the break instruction replacing a selected instruction in the object code;

processor that executes the object code such that the break instruction causes the processor to obtain and execute the hint instruction, obtain and execute the removed selected instruction, and then resume execution of the object code without changing addresses of subsequent instructions in the object code.

9. (canceled)

10. (previously presented) The computer system of claim 8, wherein the object code adapter generates a set of hint code which includes the hint instruction and the selected instruction.

11. (original) The computer system of claim 10, wherein the processor branches to the hint code when executing the break instruction.

12. (previously presented) The computer system of claim 8, wherein the processor includes a hint register for holding the hint instruction such that the processor obtains the hint instruction from the hint register and executes the hint instruction in response to the break instruction.

13. (original) The computer system of claim 12, wherein the hint register holds the selected instruction such that the processor obtains the selected instruction from the hint register and executes the selected instruction in response to the break instruction.

14. (original) The computer system of claim 12, wherein the hint register holds an address such that the processor loads the hint register using the address in response to the break instruction.

15. (original) The computer system of claim 8, wherein the object code adapter determines the hint instruction in response to a micro-architecture of the processor.

16. (previously presented) A method, comprising:

providing a sequence of instructions in object code;

replacing a selected instruction in the object code with a break instruction;

executing the break instruction to cause a processor to retrieve hint code;

executing, by the processor, the hint code, wherein the hint code includes (i) a hint instruction, (ii) the selected instruction, and (iii) an instruction to resume execution of the object code; and

resuming, upon execution of the instruction to resume, execution of the object code without changing addresses of subsequent instructions in the object code.

17. (previously presented) The method of claim 16 wherein the hint instruction includes a pre-fetch instruction that causes the processor to fetch data from memory and write the data into a cache.

18. (previously presented) The method of claim 16 wherein the hint instruction includes a branch prediction instruction.

19. (previously presented) The method of claim 16 further comprising:

providing plural hint instructions;

adapting a number of hint instructions to increase instruction execution of the processor.

20. (previously presented) The method of claim 16 further comprising:

providing plural hint instructions;

adapting a number of hint instructions depending on a cache size of the processor.

21. (previously presented) The method of claim 16 further comprising:

providing plural different types of hint instructions, wherein a type of hint instruction provided to the processor depends on functional capabilities of the processor.

22. (previously presented) The method of claim 16 wherein a break instruction replaces a selected instruction at predetermined intervals.

23. (previously presented) The method of claim 16 wherein executing the break instruction further comprises causing the processor to branch to a predetermined address having the hint code.